

### **REMARKS**

Claims 1-22 are pending in the application. Claims 1-22 stand rejected by the examiner. Assignee traverses the instant claim rejections.

#### ***Examiner's Interview***

Assignee's representatives would like to thank examiner Meng Yao Zhe for the courtesies extended to assignee's representatives (Michael Carney, Timothy Wilson, John Biernacki, and Matthew Johnson) during the telephone interview on August 16, 2007. The interview included a discussion of the specification objection on page 3 of the office action and a discussion of the Sankaran reference with respect to claim 1. In the interview, it was respectfully submitted that the Sankaran reference failed to teach agents that are servicing the request are operating synchronously with other agents and that the Sankaran reference failed to teach the interdependence between the locks as required by claim 1. Claim 2 was discussed with respect to the interdependence between the locks as being nested, and claim 18 was discussed regarding service agents constituting a legacy system. The remarks and the amendments contained herein further summarize the interview.

#### ***Claim Rejections Under 35 U.S.C. § 112 ¶2***

On page 2 of the office action, claims 1-18 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite. It is respectfully submitted that, as discussed in the interview, it is possible for the agents servicing the requests to operate synchronously with respect to each other in order to service different asynchronized tasks that are making the request. For example, the set of servicing agents could be a group of task-

based code modules (e.g., I/O request handlers, etc.). In contrast to threads which operate in a multi-threaded environment, task-based code operates in a single threaded environment. In general, more than one thread can be operating at the same time, but more than one task cannot be executing at the same time. More specifically, a task is a cooperatively scheduled entity, and only one task can run at a time in a processor. A task usually waits to run until it is implicitly scheduled by another task yielding control of the process. A thread is a preemptively scheduled entity, whose execution is controlled in large part by the scheduling mechanisms of the native operating system. Through the aspects detailed in the disclosure, the function of the synchronous task-based code modules can be coordinated with the asynchronous threads requesting the task-based modules' services. (See paragraph [0018] of the published instant application). Accordingly, it is respectfully submitted that it is clear how the cited claim feature can be accomplished, and it is respectfully requested that the rejection under 35 U.S.C. 112 be withdrawn.

### ***Specification Objection***

On page 3 of the office action, the specification is objected to for failing to support the feature of the agents (that are servicing the request) are operating synchronously with respect to each other. In view of the above discussion that is based upon assignee's specification (e.g., paragraph [0018] of the published instant application), it is respectfully submitted that the specification does support the cited feature, and it is respectfully requested that the specification objection be withdrawn.

### *Claim Rejections Under 35 U.S.C. § 102(e)*

Claims 1-22 stand rejected under 35 U.S.C. 102(e) as being anticipated by the Sankaran reference. Assignee respectfully disagrees. For example, it is respectfully submitted that the Sankaran reference does not render claim 1 unpatentable. Claim 1 recites a dispatch lock for allowing only one requesting thread into a dispatch section at a time, and a service pool lock for synchronizing the requesting thread that is in the dispatch section with a service agent. As recited in claim 1, the service agents servicing the requests operate synchronously with respect to each other.

The Sankaran reference fails to disclose such limitations. For example, the Sankaran reference makes no reference to synchronized operation of service agents in its disclosure especially in relation to the purpose recited in claim 1 for the service agents. It also does not discuss the handling of requests for synchronously operating service agents by asynchronously operating requesting threads. Therefore, it is respectfully submitted that the rejection of independent claim 1 based upon the Sankaran reference is improper because the reference fails to teach all of the claimed features recited in claim 1. Independent claims 21 and 22 contain similar language as independent claim 1, and it is respectfully submitted that these claims are in condition for allowance based on similar reasoning as discussed for independent claim 1 in this paragraph.

Additionally, the Sankaran reference fails to teach the final claimed feature of claim 1, wherein after the requesting thread releases the first and second lock, the service agent handles the request of the requesting thread. While the Sankaran reference may discuss two locks, there is no disclosure in the Sankaran reference that any service agent is handling the requests following release of the two locks. In fact, it is respectfully

submitted that just the opposite occurs because any reading/writing operations which would require seeking a lock in the Sankaran reference would keep that lock in place while processing the operation (e.g., a lock sought for updating a data record would be kept in place until the write is complete to avoid access to that data record during writing). This is in direct contradiction to the language of claim 1 of the instant application which recites that both the first and second locks are released prior to handling the request of the requesting thread. Because the Sankaran reference fails to teach this feature, it is respectfully submitted that the rejection of claim 1 under 35 U.S.C. § 102(e) is improper, and it is requested that this rejection be withdrawn. Independent claims 19 and 21 contain similar language as independent claim 1 concerning this claimed feature, and it is respectfully submitted that these claims are in condition for allowance based on similar reasoning as discussed for independent claim 1 in this paragraph.

As discussed in the interview, claim 1 has been amended to recite that the first lock is operating as a dispatch lock and the second lock is operating as a service pool lock. Also as discussed in the interview, claim 2 has been amended to clarify the term “nested,” and claim 18 has been amended to further describe the meaning of a legacy system. Thus, it is respectfully submitted that these dependent claims are in condition for allowance.

Assignee at this time has not submitted any arguments in support of the patentability of the dependent claims not mentioned above. It is believed that independent claims 1, 19, 21, and 22 are now in condition for allowance such that all of

the dependent claims which depend either directly or indirectly therefrom are also in condition for allowance.

### CONCLUSION

For the foregoing reasons, assignee respectfully submits that the pending claims are allowable. Therefore, the examiner is respectfully requested to pass this case to issue.

Respectfully submitted,

By: 

John V. Biernacki  
Reg. No. 40,511  
JONES DAY  
North Point  
901 Lakeside Avenue  
Cleveland, Ohio 44114  
(216) 586-3939